Amendments to the Claims:

This listing of claims will replace all prior versions, and listings, of claims in the application:

Listing of Claims:

1. (Currently amended) A method of identifying false cache line sharing during execution of a computer program in a multiprocessor data processing system, comprising:

associating processor flag bits with at least one portion of a cache line in a cache, wherein the processor flag bits include a processor flag bit for each processor of the multiprocessor data processing system;

determining whether the cache line is being falsely shared between processors based on values of the processor flag bits; and

modifying storage of [[the]] a contents of the cache line to avoid false sharing of the cache line based on a determination that the cache line is being falsely shared between processors of the multiprocessor data processing system.

2. (Original) The method of claim 1, wherein determining whether the cache line is being falsely shared between processors further includes:

sending an interrupt to an interrupt handler in response to a determination that the cache line is being falsely shared between processors.

3. (Currently amended) The method of claim 2, wherein modifying storage of the contents of the cache line to avoid false sharing of the cache line includes:

storing contents of a portion of the cache line in a separate memory area; and

modifying [[the]] code of the computer program to redirect access requests to the portion of the cache line to the separate memory area.

4. (Original) The method of claim 1, further comprising:

receiving an access request directed to a portion of the cache line; and

sending an interrupt to an interrupt handler associated with a performance monitoring application in response to a determination that false cache line sharing between processors is present with regard to the cache line.

5. (Currently amended) The method of claim 1, wherein determining whether the cache line is being falsely shared between processors includes:

reading values of the processor flag bits for each portion of the cache line in response to a reload of the cache line due to an access operation from a current processor of a plurality of processors, in the multiprocessor data processing system, targeting a current portion of the cache line; and

determining if the reload ~~operation~~ is due to false sharing of the cache line based on the values of the processor flag bits for each portion of the cache line.

6. (Currently amended) The method of claim 5, wherein determining if the reload ~~operation~~ is due to false sharing of the cache line includes:

comparing a [[first]] value of a first processor flag bit associated with the current processor and the current portion to values of second processor flag bits associated with other processors of the plurality of processors and the current portion; and

determining that true cache line sharing is present if a first predetermined relationship between the value of the first processor flag bit and the values of the second processor flag bits is present.

7. (Currently amended) The method of claim 6, further comprising:

comparing the [[first]] value of the first processor flag bit with values of third processor flag bits associated with other processors and another portion of the cache line; and

determining that false cache line sharing is present if a second predetermined relationship between the value of the first processor flag bit, the values of the second processor flag bits, and the values of the third processor flag bits is present.

8. (Original) The method of claim 6, wherein the first predetermined relationship is that at least one of the second processor flag bits is set and the first processor flag bit is set.

9. (Original) The method of claim 7, wherein the second predetermined relationship is that at least one of the third processor flag bits is set, a corresponding one of the second processor flag bits is not set, and the first processor flag bit is set.

10. (Currently amended) The method of claim 5, further comprising:

outputting an indication that there is false cache line sharing if the reload ~~operation~~ is determined to be due to false sharing of the cache line.

11. (Currently amended) A computer program product in a recordable-type computer readable medium for identifying false cache line sharing during execution of a computer program in a multiprocessor data processing system, comprising:

first instructions for associating processor flag bits with at least one portion of a cache line in a cache, wherein the processor flag bits include a processor flag bit for each processor of the multiprocessor data processing system;

second instructions for determining whether the cache line is being falsely shared between processors based on values of the processor flag bits; and

third instructions for modifying storage of [[the]] a contents of the cache line to avoid false sharing of the cache line based on a determination that the cache line is being falsely shared between processors of the multiprocessor data processing system.

12. (Original) The computer program product of claim 11, wherein determining whether the cache line is being falsely shared between processors further includes:

fourth instructions for sending an interrupt to an interrupt handler in response to a determination that the cache line is being falsely shared between processors.

13. (Currently amended) The computer program product of claim 12, wherein the third instructions for modifying storage of the contents of the cache line to avoid false sharing of the cache line include:

instructions for storing contents of a portion of the cache line in a separate memory area; and

instructions for modifying [[the]] code of the computer program to redirect access requests to the portion of the cache line to the separate memory area.

14. (Original) The computer program product of claim 11, further comprising:

fourth instructions for receiving an access request directed to a portion of the cache line; and

fifth instructions for sending an interrupt to an interrupt handler associated with a performance monitoring application in response to a determination that false cache line sharing between processors is present with regard to the cache line.

15. (Currently amended) The computer program product of claim 11, wherein the second instructions for determining whether the cache line is being falsely shared between processors include:

instructions for reading values of the processor flag bits for each portion of the cache line in response to a reload of the cache line due to an access operation from a current processor of a plurality of processors, in the multiprocessor data processing system, targeting a current portion of the cache line; and

instructions for determining if the reload ~~operation~~ is due to false sharing of the cache line based on the values of the processor flag bits for each portion of the cache line.

16. (Currently amended) The computer program product of claim 15, wherein the instructions for determining if the reload ~~operation~~ is due to false sharing of the cache line include:

instructions for comparing a [[first]] value of a first processor flag bit associated with the current processor and the current portion to values of second processor flag bits associated with other processors of the plurality of processors and the current portion; and

instructions for determining that true cache line sharing is present if a first predetermined relationship between the value of the first processor flag bit and the values of the second processor flag bits is present.

17. (Currently amended) The computer program product of claim 16, further comprising:

instructions for comparing the [[first]] value of the first processor flag bit with values of third processor flag bits associated with other processors and another portion of the cache line; and

instructions for determining that false cache line sharing is present if a second predetermined relationship between the value of the first processor flag bit, the values of the second processor flag bits, and the values of the third processor flag bits is present.

18. (Original) The computer program product of claim 16, wherein the first predetermined relationship is that at least one of the second processor flag bits is set and the first processor flag bit is set.

19. (Original) The computer program product of claim 17, wherein the second predetermined relationship is that at least one of the third processor flag bits is set, a corresponding one of the second processor flag bits is not set, and the first processor flag bit is set.

20. (Currently amended) The computer program product of claim 15, further comprising:

fourth instructions for outputting an indication that there is false cache line sharing if the reload ~~operation~~ is determined to be due to false sharing of the cache line.

21. (Currently amended) An apparatus for identifying false cache line sharing during execution of a computer program in a multiprocessor data processing system, comprising:

means for associating processor flag bits with at least one portion of a cache line in a cache, wherein the processor flag bits include a processor flag bit for each processor of the multiprocessor data processing system;

means for determining whether the cache line is being falsely shared between processors based on values of the processor flag bits; and

means for modifying storage of [[the]] a contents of the cache line to avoid false sharing of the cache line based on a determination that the cache line is being falsely shared between processors of the multiprocessor data processing system.